OBJECT ORIENTED PROGRAMMING

Submitted To:

Mam Hira Saleem

Submitted By:

Nayab Fatima

Roll No:

2K23-BSCS-612

Section:

C

Final Project:

Chatbot

NFC INSTITUTE OF ENGINEERING & TECHNOLOGY
FEDERAL DEGREE AWARDING INSTITUTE
NFC IET
MULTAN - PAKISTAN

SOURCE CODE:

ChatBotProgramm - Notepad

File Edit Format View Help

```
import java.util.Scanner;
// Interface for response
interface ResponseGenerator {
    String generateResponse(String input);
}

// Abstract class
abstract class ChatBot {
    protected String name;

    // Constructor
    public ChatBot(String name) {
        this.name = name; // instance variable name of the ChatBot object
    }

    public abstract void introduce(); // abstract methods

    public abstract void chat();
}

// Predefined messages that bot responds to
class SimpleChatBot extends ChatBot {
    private ResponseGenerator responseGenerator;

    public SimpleChatBot(String name, ResponseGenerator responseGenerator) {
        super(name);
        this.responseGenerator = responseGenerator;
    }
```

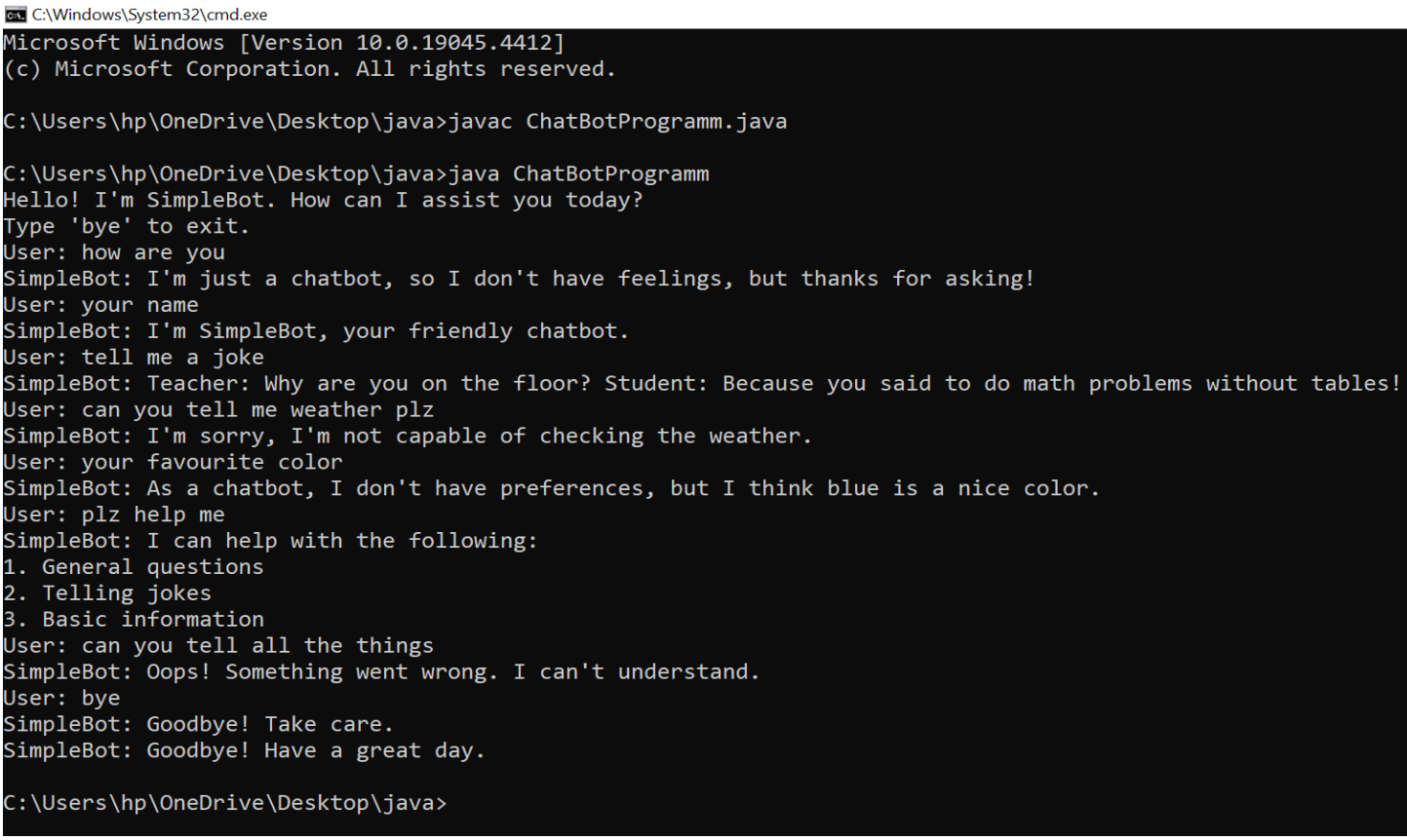
ChatBotProgramm - Notepad

File Edit Format View Help

```
    public void introduce() {
        System.out.println("Hello! I'm " + name + ". How can I assist you today?");
        System.out.println("Type 'bye' to exit.");
    }

    public void chat() {
        Scanner scanner = new Scanner(System.in); // receive input from the user.
        String userInput;

        do {
            System.out.print("User: ");
            userInput = scanner.nextLine().toLowerCase().trim();

            String botResponse = responseGenerator.generateResponse(userInput);
            System.out.println(name + ": " + botResponse);

        } while (!userInput.equals("bye"));

        System.out.println(name + ": Goodbye! Have a great day.");
        scanner.close();
    }
}

// Main class
public class ChatBotProgramm {
    public static void main(String[] args) {
        ResponseGenerator responseGenerator = new ResponseGenerator() {
            // Implement the method of interface ResponseGenerator
            public String generateResponse(String input) {
```

```
    // Implement the method of interface ResponseGenerator
    public String generateResponse(String input) {
        if (input.contains("how are you")) {
            return "I'm just a chatbot, so I don't have feelings, but thanks for asking!";
     } else if (input.contains("weather")) {
            return "I'm sorry, I'm not capable of checking the weather.";
    } else if (input.contains("joke")) {
        return "Teacher: Why are you on the floor? Student: Because you said to do math problems without tables!";
    } else if (input.contains("bye")) {
            return "Goodbye! Take care.";
    } else if (input.contains("your name")) {
            return "I'm SimpleBot, your friendly chatbot.";
    } else if (input.contains("time")) {
            return "I'm sorry, I cannot tell the time.";
    } else if (input.contains("help")) {
            return "I can help with the following: \n1. General questions\n2. Telling jokes\n3. Basic information";
    } else if (input.contains("day")) {
            return "I hope you're having a wonderful day!";
    } else if (input.contains("favorite color") || input.contains("favourite color")) {
            return "As a chatbot, I don't have preferences, but I think blue is a nice color.";
    } else {
            return "Oops! Something went wrong. I can't understand.";
        } } };

  ChatBot simpleChatBot = new SimpleChatBot("SimpleBot", responseGenerator);
// Constructor call
simpleChatBot.introduce();
simpleChatBot.chat();
  } }
```

Output:



Explanation:

Package:

Chatbot is actually an AI app which can chat with user and to read user input by SCANNER package that import in this code.

Interface: Response Generator

In this code an interface is used to generate response like response generated interface.in this interface a method is

used. A class which implements the interface must implement their method. This interface is used user input as input and give respective response as an output. In the end main class implement this interface and implements their methods which give response as a return of this interface.

Abstract Class: Chatbot

This abstract class represents a chatbot with two abstract methods (introduce and chat). The name attribute is used to store the name of the chatbot.

Constructor: Initializes the name attribute of the chatbot.

Inheritance:

A class extends from its parent class. In second picture First two steps is implementation of the abstract method are done. Next a do-while loop is used which continue to execute value of user input till input is not equal to **"bye"**.

Chat ():

- Uses a Scanner to read user input.
- Calls the generate Response method of the response Generator to get a response based on the user input.
- Calls the generate Response method of the response Generator to get a response based on the user input.
- Prints the response.
- Loops until the user types "bye".
- Closes the Scanner and prints a goodbye message.

Main Class:

And in the last main class is created. And then if-else condition is used. Statement of if checks if the user's input contains the phrase **"how are you"**. If it does, the chatbot responds with a predefined message which I write in code. Otherwise else condition will execute. In this class interface is implemented and their method is must implemented which give output as their response. And at the end we call introduce Bot and chat method.

Summary:

The program defines a simple chatbot framework where the chatbot can respond to specific user inputs with predefined messages. The Simple Chatbot class handles user interaction by reading input, generating appropriate responses using the Response Generator interface, and printing those responses to the user. The main class (Chatbot Programm) initializes and runs the chatbot.